REPORT DOCUMENTATION PAGE

Form Approved
OMB NO. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE

3. REPORT TYPE AND DATES COVERED

~~Reprint~~

4. TITLE AND SUBTITLE

TITLE ON ~~REPRINT~~

5. FUNDING NUMBERS

DAAG55-98-1-0230

6. AUTHOR(S)

AUTHOR(S) ON ~~REPRINT~~

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(ES)

University of Texas - Austin

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

U.S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

ARO 37634.19-PH

11. SUPPLEMENTARY NOTES

The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision, unless so designated by other documentation.

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12 b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

ABSTRACT ON ~~REPRINT~~

20011024 055

14. SUBJECT TERMS

15. NUMBER IF PAGES

16. PRICE CODE

17. SECURITY CLASSIFICATION OR REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

Segmentation and Recognition of Continuous Human Activity *

Anjum Ali and J. K. Aggarwal
Computer and Vision Research Center
Department of Electrical and Computer Engineering
The University of Texas at Austin, TX 78712, USA
aggarwaljk@mail.utexas.edu

Abstract

This paper presents a methodology for automatic segmentation and recognition of continuous human activity. We segment a continuous human activity into separate actions and correctly identify each action. The camera views the subject from the lateral view. There are no distinct breaks or pauses between the execution of different actions. We have no prior knowledge about the commencement or termination of each action. We compute the angles subtended by three major components of the body with the vertical axis, namely the torso, the upper component of the leg and the lower component of the leg. Using these three angles as a feature vector, we classify frames into breakpoint and non-breakpoint frames. Breakpoints indicate an action's commencement or termination. We use single action sequences for the training data set. The test sequences, on the other hand, are continuous sequences of human activity that consist of three or more actions in succession. The system has been tested on continuous activity sequences containing actions such as walking, sitting down, standing up, bending, getting up, squatting and rising. It detects the breakpoints and classifies the actions between them.

1 Introduction

Human activity is a continuous flow of single or discrete human action primitives in succession. An example of a human activity is a sequence of actions in which a subject enters a room, sits down, then stands up, walks forward, bends down to pick up something, and then gets up and walks away. Each component of the human activity, such as walking, sitting down, standing up, bending down and getting up, is a discrete action primitive. Methodology for automatic interpretation of such continuous activity is presented in this paper. When humans move from one action to another, they do so smoothly; transitions between actions are not clearly defined. In general, there is no clear beginning or end of an action. Therefore, to recognize a continuous activity sequence such as the one described above, the detection of transitions between actions is crucial. Most human activity recognition systems require the input sequence to be a single action sequence. In other cases, systems recognize the poses associated with an action rather than a complete action. This enables them to recognize actions but not necessarily to give an accurate temporal description of each action. In this paper, we analyze continuous human activity by first automatically segmenting the activity into discrete actions. Human body motion is actually the movement of the body's parts or components, such as the torso or the upper and the lower limbs. We find that these components, the torso and upper and lower limbs, are the most informative in identifying 'breakpoints' between the actions that we aim to segment and recognize. The system gives an output of all actions that have taken place during the course of a sequence and their individual time intervals in terms of time frames.

2 Review of Previous Work

Most research in the area of human activity recognition has dealt with the recognition of discrete action primitives. Segmentation and classification of continuous actions is virtually unexplored. The system presented by Madabhushi and Aggarwal [7] classifies twelve different classes of actions. These actions are walking, sitting, standing up, bending, getting up, bending sideways, falling, squatting, rising and hugging in the frontal or lateral views. They track the movement of the head over successive frames and model their system using the difference in the coordinates of the head. They achieved a recognition rate of 83 percent. However, each test sequence was a discrete action primitive. Bobick and Davis [4] used temporal templates for the representation and recognition of human actions. They classified 18 aerobic exercises in 7 different orientations. Once again, the approach was to recognize discrete actions, which in their case were aerobic exercises. Ayers and Shah [2] pre-

*This research was supported in part by the Army Research Office under contracts DAAH04-95-I-0494 and DAAG55-98-1-0230, and by the Texas Higher Education Coordinating Board, Advanced Research Project 97-ARP-275.

sented a context-based action recognition system capable of determining the actions taking place in a room. It recognized actions such as walking into a room, opening a cabinet, picking up a telephone and using a computer terminal. Although their system gave a rundown of all of the actions taking place in a room, it was not able to detect exactly when a subject proceeds from one action to the next. Campbell and Bobick [3] presented a system to recognize continuous actions in a limited context. Their system recognized nine fundamental ballet steps from three-dimensional point data. They were thus not using video information but 3D data as input. They used a set of anatomical constraints to model each step. They were not detecting transitions between steps; whenever a certain set of constraints was observed somewhere during the course of a sequence, the system labelled the corresponding step. A more recent related work is the one by Rui and Anandan [10]. In this the authors segment visual action sequences based on detecting temporal discontinuities in spatial motion patterns. They extract the frame by frame optical flow and using singlular value decomposition detect discontinuities in trajectories. These discontinuities are *keypose frames* that are the boundaries of actions.

Human body motion is the coordinated movement of different body parts and the connected joints. We believe that knowledge of limb and joint angles is useful in detecting the termination and commencement of different actions. A number of studies have used information from the movement of body parts such as the trunk, arms and legs to analyze human motion. Rohr [9] described human walking with joint angles of the hip, knee, shoulder and elbow. In the same vein, Bharatkumar *et al* [1] used kinesiology data as the basis for their human walking model. Fujiyoshi and Lipton [5] used the angle of inclination of the torso as a cue to the recognition of walking and running. Niyogi and Adelson [8] exploited the repetitive information of the lower limb trajectory for recognition of human walking.

In this paper we present an algorithm that uses the angle of inclination of three major body components to classify frames into breakpoint and non-breakpoint frames. We then classify the frames between the breakpoints into one of the actions present in the database. The organization of this paper is as follows. In section 3 we present the preprocessing steps applied to the images in order to extract the features. Section 4 explains the algorithm for action segmentation, with a detailed description of each step and some examples. Section 5 describes the module for discrete action recognition, enumerating the features used for classification of the individual actions. System implementation and results are presented in section 6. Conclusions and future directions are outlined in section 7.

3 Preprocessing steps: Segmentation and skeletonization

The accurate segmentation of the subject in each frame of the sequence is critical to the skeletonization process, which is sensitive to boundary and internal discontinuities. Background subtraction is used to segment the subject from the scene. This is followed by thresholding, which yields a binary image. The resulting image is further processed using morphological operations such as dilation, erosion and connected component labeling. Fig.1 shows the segmentation result for one frame of a sequence.

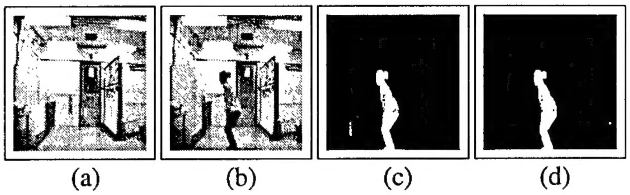

Figure 1: (a) background image (b) one frame of a sequence (c) thresholded image (d) final segmented image

Skeletonization has been used extensively in human motion analysis to extract a skeletonized image of the human subject or to generate stick figure models. Bharatkumar *et al.* [1] used the medial axial transformation to extract stick figures and compared the two-dimensional information obtained from stick figures with that obtained from anthropometric data. Guo *et al.* [6] also used skeletonization on the extracted human silhouette to yield stick figure models. Since we are working with actions in the lateral view, skeletonization can be used to obtain the three main components of the body used in our algorithm, namely, the torso, the upper component of the legs and the lower component of the legs. Prior work has used a priori information about the position of the hip and the knee joints.

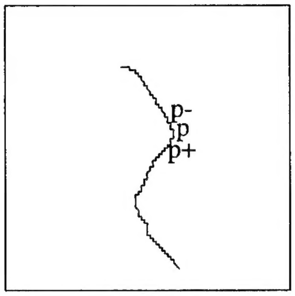

Figure 2: Configuration of points on the skeleton curve

The hip and knee regions are detected by estimating the highest points of curvature on the skeleton. Regions of high

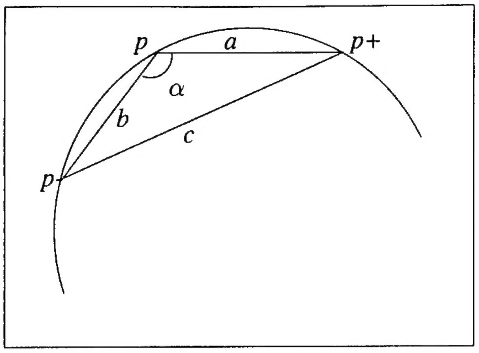

Figure 3: Computation of the angle of curvature α

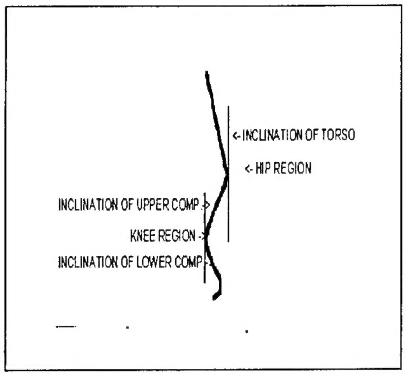

Figure 4: Skeletonization result of image frame in Fig.1.

curvature will in turn subtend small angles along the curve. The skeleton is represented by a sequence of points p_i in the image plane. For each point p_i along the skeleton curve, we compute the opening angle α using the following formula

$$\alpha = \arccos((a^2 + b^2 - c^2)/2 \times a \times b) \qquad (1)$$

where a, b, and c are distances computed as $|p - p^+| = |a|$, $|p - p^-| = |b|$ and $|p^+ - p^-| = |c|$. p^+ and p^- are points on the skeleton curve that are on either side of p (Fig.2 and Fig.3) and within a specified pixel range δ. δ is set to 1/10th the height of the skeleton in the present frame. This is to take into account the change in the height of the skeleton as the subject performs different actions. Thus for every point p along the curve, we find the angle subtended at that point by using two other points that are above and below p at a pixel distance δ. Note that δ is measured in pixels and a,b and c are absolute distances between the points. We then find points that subtend minimal angles along the curve. These are regions of high curvature. The hip is the first point of high curvature. The knee is then the point of high curvature that occurs below the hip. Once the hip and the knee regions are detected, the angles are computed with respect to the vertical axis passing through the hip and the knee.

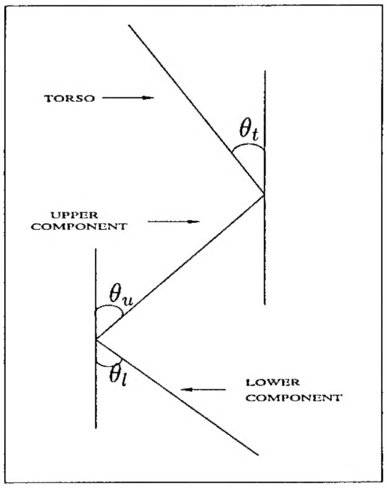

Figure 5: Schematic representation of the body components and associated angles

Fig.4 shows the three components of the skeleton for the sequence frame depicted in Fig.1. Fig.5 gives the corresponding schematic representation of the body components and associated angles.

4 Algorithm for action segmentation

The algorithm for action segmentation uses the angle of inclination of the torso, denoted by θ_t, the angle of inclination of the upper component of the legs, θ_u, and the angle of inclination of the lower component of the legs, θ_l. These three angles form a feature vector $\{\theta_t, \theta_u, \theta_l\}$. The steps of the algorithm are given below.

4.1 Computation of component angles

For each frame of the test sequence, the algorithm computes the three angles of inclination of the body components. During a continuous activity sequence, θ_t, θ_u and θ_l traverses a series of maximas and minimas. Fig. 6 plots θ_t, θ_u and θ_l respectively for a the sample test sequence illustrated in Fig.13.

4.2 Classification of frames into breakpoint frames

Each frame of a continuous sequence is represented by a feature vector, $\{\theta_t$, θ_u ,$\theta_l\}$. We define two classes, a breakpoint class and a non-breakpoint class. Training vectors for both classes are chosen from continuous sequences. Frames

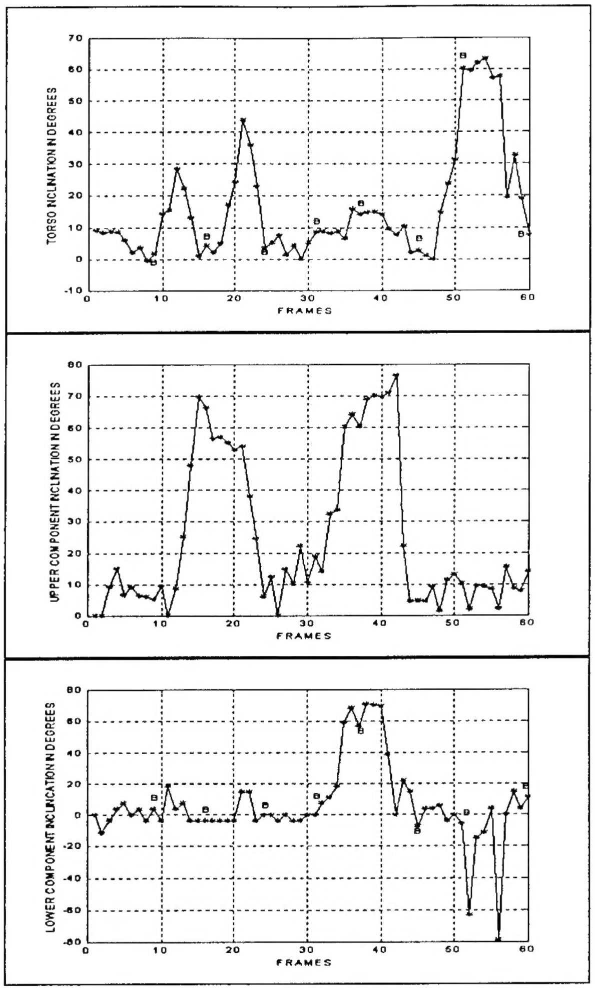

Figure 6: Angle of inclination of the torso, the upper component and the lower component for the sample test sequence.

of a continuous sequence in which a person is at the commencement or the termination of an action are chosen to form the breakpoint class. Each sample of this class is represented by a three-element feature vector represented by $\{\theta_{tb}, \theta_{ub}, \theta_{lb}\}$. Fig.7 shows some of the training sample frames that have been used for the breakpoint class. 16 sample frames were used in the training.

Frames in which the subject is in the middle of executing an action are chosen to form the non-breakpoint class. Each sample of this class is represented by $\{\theta_{tr}, \theta_{ur}, \theta_{lr}\}$. Fig.8 shows some of the training sample frames that were used for the non-breakpoint class. 28 sample frames were used to train the non-breakpoint class. The three-element test feature vector $\{\theta_t, \theta_u, \theta_l\}$ is compared with the training feature vectors from each class. The algorithm computes two Euclidean distance measures D_b and D_r between the feature vector for each frame of a test sequence and the training feature vectors of the breakpoint and non-breakpoint classes respectively. A frame is classified as a breakpoint frame if the minima of the distance measure set D_b is less than the minima of measure set D_r. We are basically looking for combinations of component angles that characterize a breakpoint between actions. This becomes clearer if we look at the graphs in Fig. 6, which indicate the frames that were detected as breakpoint frames for the sequence in Fig.13. We observe that breakpoints have been detected at frames that are transitions between actions.

Figure 7: Samples of breakpoint training frames

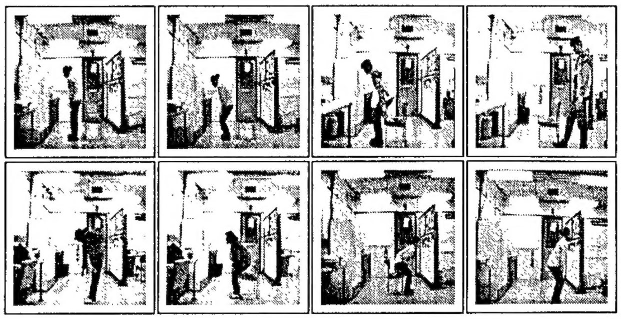

Figure 8: Samples of non-breakpoint training frames for different sequences

4.3 Segmenting individual actions from a continuous sequence

Frames that lie between breakpoint frames are segmented as individual actions. The algorithm initiates a counter every time a breakpoint frame is detected. The algorithm then keeps track of frames and looks for the next breakpoint frame. Once the next breakpoint is detected, frames between the two breakpoints are classified as an individual action. Sometimes more than one frame in the vicinity of the breakpoint frame can get classified as a breakpoint frame. In this case, we pick the breakpoint which yields the smallest value of D_b.

5 Discrete action recognition

In addition to breakpoint detection, we use the angle of inclination of the torso and the upper component and lower components to classify the different actions. We observe

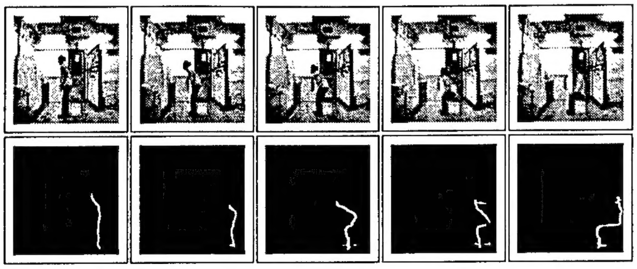

Figure 9: Frames of a sitting down action and its skeleton showing the components

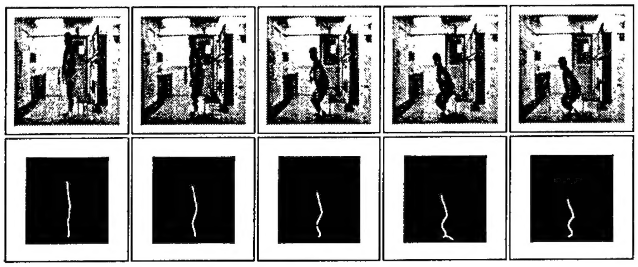

Figure 10: Frames of a squatting action and its skeleton showing the components

that these angles traverse a characteristic path during the execution of each action. The skeletons for sitting down and squatting are shown in Fig.9 and Fig.10 respectively. The feature vectors can be given as follows:

$$A1 = [\theta_{t1}, \theta_{t2}, \ldots, \theta_{tn}] \tag{2}$$

$$A2 = [\theta_{u1}, \theta_{u2}, \ldots, \theta_{un}] \tag{3}$$

$$A3 = [\theta_{l1}, \theta_{l2}, \ldots, \theta_{ln}] \tag{4}$$

where $A1$, $A2$, $A3$ are the normalized vectors for the angle of inclination of the torso, the upper component and the lower component of the leg respectively. The angles are normalized for each action by dividing them with the maxima of the absolute values for that angle. The system has been trained on complete discrete action sequences that last for ten frames. If the test action yields a feature vector with fewer elements than the training vectors, it is interpolated to the size of the training vector. Similarly, if the test vector is longer than the training vector, then all the training vectors are interpolated to the length of the test vector. All three feature vectors are interpolated using cubic spline interpolation.

The nearest neighbor classifier assigns the feature vector $\{A1, A2, A3\}$ to the same class Ω_ω (where $\omega \in \{1, 2, \ldots, 7\}$) as the training feature vectors nearest to it in the feature space. The test sequence is assigned to the training class that yields the least sum as computed in Eq. 5.

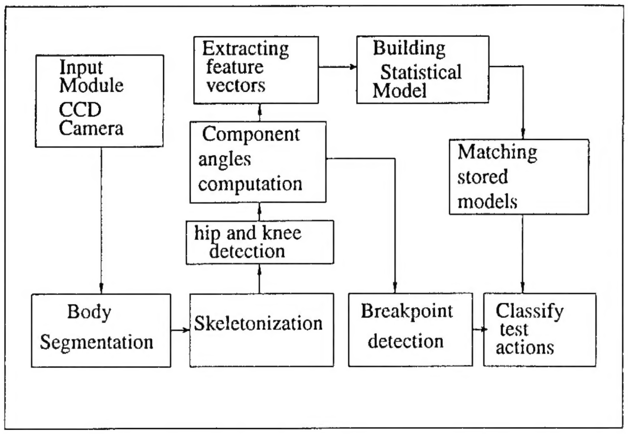

Figure 11: System Implementation

$$\min_{\omega}\{\sum_{k=1}^{n}(\theta_{tk} - \theta_{t\omega k})^2 + \sum_{k=1}^{n}(\theta_{uk} - \theta_{u\omega k})^2 + \sum_{k=1}^{n}(\theta_{lk} - \theta_{l\omega k})^2\} \tag{5}$$

6 System Implementation and Results

Image sequences are obtained using a fixed CCD camera working at 12-15 frames per seconds. The test sequences range in size from 60 to 80 frames. Figure 11 illustrates the different system modules. The system segments the body of the subject. After the body segmentation, the resulting image is skeletonized and the component angles are extracted. The system then segments the sequence into actions using the detected breakpoints. Each action is then classified using the nearest neighbor classifier. The algorithm has been tested on 20 sequences of continuous activity. The test and training sequences are different and 6 subjects have been used in this work. Each test sequence consists of actions performed in a continuous manner with no breaks or pauses. The 20 sequences contain, in all, 143 actions and 128 breakpoints. Table 1 gives the results that have been obtained on the test sequences. The results demonstrate the efficiency of the algorithm with respect to breakpoint detection and action recognition. Figure 13 contains frames 1 to 59 of test sequence 1. Table 2 gives the results for this sequence. When the breakpoint is not correctly identified, the probability of incorrect action classification is higher.

Number of :	Total	Correct	Efficiency
Breakpoints	128	110	85.90
Actions	143	110	76.92
Walking	29	26	89.66
Sitting	13	10	76.92
Standing up	16	12	75.00
Bending	21	15	71.42
Getting up	19	14	73.68
Squatting	24	18	75.00
Rising	21	15	71.42

Table 1: Results for test sequences

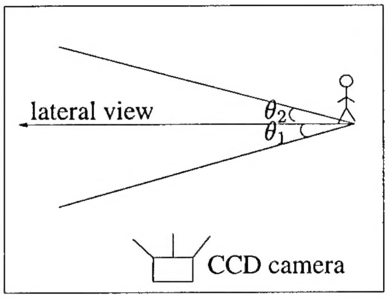

Figure 12: Set up

7 Conclusion

We have presented a methodology for automatic segmentation and recognition of continuous human activity. The activity sequence consists of actions that are performed in succession without any breaks or pauses. The sequence is segmented into individual action primitives. The system uses the angles subtended by three major components of the body to classify frames of the sequence into breakpoint and non-breakpoint frames. These components are the torso, the upper leg and the lower leg. The action between two breakpoint frames is then classified using a three-element feature vector. Although the system is limited to the lateral viewpoint of the body, we have tested it with sequences in which the camera does not view the subject from a perfect lateral view. Different angles of inclinations with the plane of the camera have been tested. The system can tolerate a deviation of 40 degrees in θ_1 and 25 degrees in θ_2 as illustrated in Fig.12. We have also tested on sequences in which θ_1 and θ_2 change during the execution of an activity. Figure 14 is a sequence in which the camera views the subject at angle θ_1 = 40 degrees. The results of this sequence are in Table 3. The sequence has not been numbered due to page restrictions. The system starts to fail when the high points of curvature on the skeleton contour are no longer discernible in the field of view. False hip and knee regions also get generated in certain cases. We would like to find a more efficient method of detecting the hip and knee regions under these conditions. Further tracking the trajectory of other parts of the body along with the leg components could make the system more robust. This would help in recognizing more complex actions taken from different views.

References

[1] A.G.Bharatkumar, K.E. Daigle, M.G. Pandy, Qin Cai, and J.K. Aggarwal. Lower limb kinetics of human walking with the medial axis transformation. *In Proceedings IEEE Workshop on Motion of Non-Rigid and Articulated Objects*, pages 70–76, 1994. Austin, Texas.

[2] Douglas Ayers and Mubarak Shah. Recognizing human actions in a static room. *In Proceedings Applications of Computer Vision, WACV'98, Fourth IEEE Workshop*, pages 42–47, 1998. Princeton, New Jersey.

[3] Lee Campbell and Aaron Bobick. The recognition of human body motion using phase space constraints. *In Proceedings Fifth International Conference on Computer Vision*, pages 624–630, 1995. Cambridge, Massachusetts.

[4] James W. Davis and Aaron F. Bobick. The representation and recognition of human movement using temporal templates. *In Proceedings Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, pages 928–934, 1997. Venice, Italy.

[5] H. Fujiyoshi and Alan J. Lipton. Real time human motion analysis by image skeletonization. *Fourth IEEE Workshop on Applications of Computer Vision, WACV*, pages 15–21, 1998. Princeton, New Jersey.

[6] Yan Guo, Gang Xu, and Saburo Tsuji. Understanding human motion patterns. *Proceedings of the 12th IAPR Conference on Computer Vision and Image Processing*, pages 325–329, 1994. Jerusalem, Israel.

[7] Anant Madabhushi and J. K. Aggarwal. Using head movement to recognize human activity. *15th International Conference on Pattern Recogntion*, pages 698–701, 2000. Baracelona, Spain.

[8] S. A. Niyogi and E. H. Adelson. Analyzing and recognizing walking figures in xyt. *IEEE Computer Society's Conference on Computer Vision and Pattern Recognition*, pages 469–474, 1994. Seattle, Washington.

[9] K. Rohr. Towards model-based recognition of human movements in image sequences. *CVGIP Image Understanding*, 59(1):94–115, 1994.

[10] Yong Rui and P. Anandan. Segmenting visual actions based on spatio-temporal motion patterns. *IEEE Computer Society's Conference on Computer Vision and Pattern Recognition*, pages 111–118, 2000. Hilton Head Island, South Carolina.

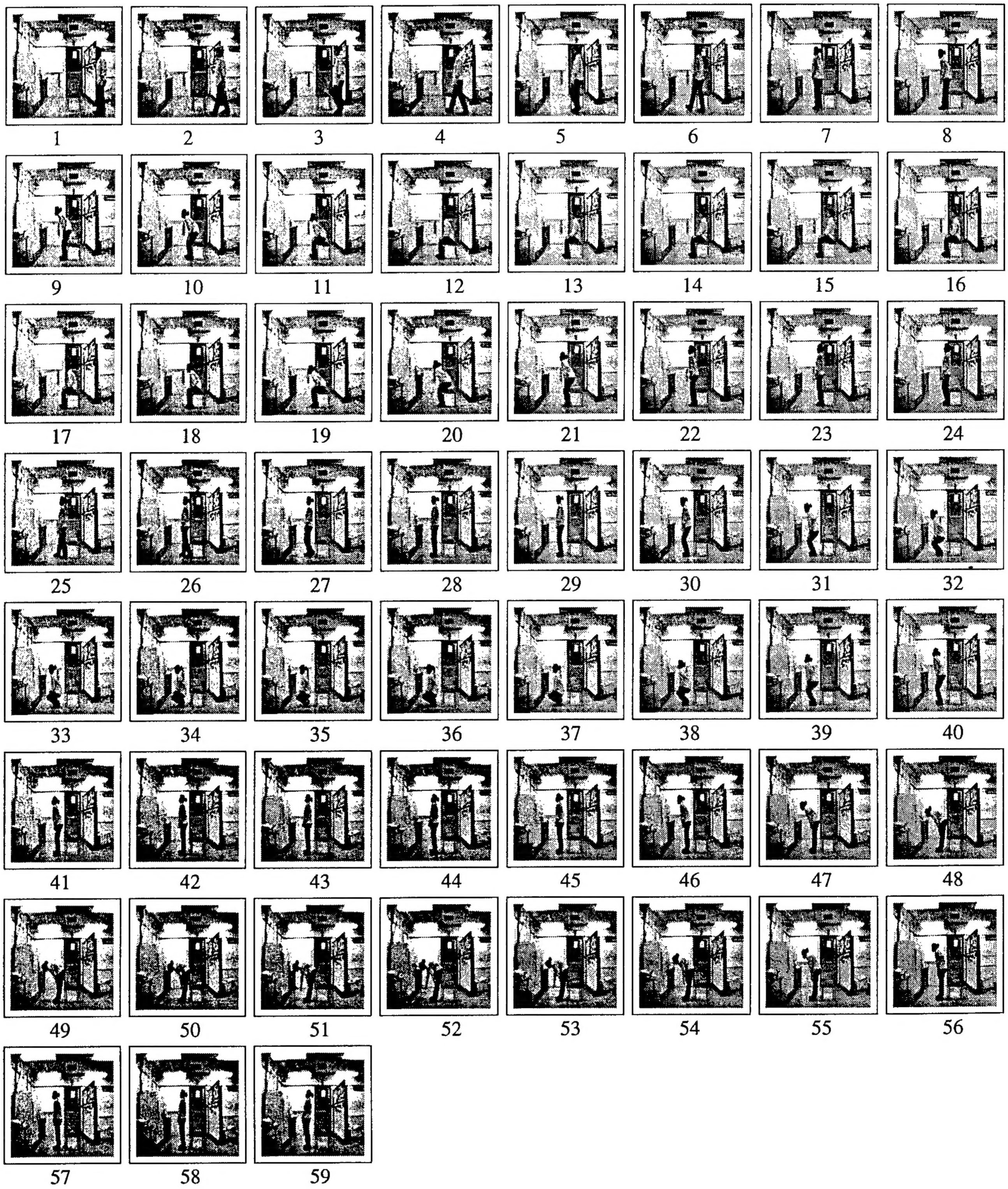

Figure 13: Frames 1 to 59 of test sequence 1

Test Sequence 1								
Actual frames	1-8	8-17	17-25	25-30	30-37	37-43	43-52	52-59
action	walking	sitting	standingup	walking	squatting	rising	bending	getting up
Detected frames	1-8	9-16	16-24	25-30	31-37	38-45	45-51	51-59
action	walking	sitting	standingup	walking	squatting	rising	bending	getting up

Table 2: Results for test sequence 1

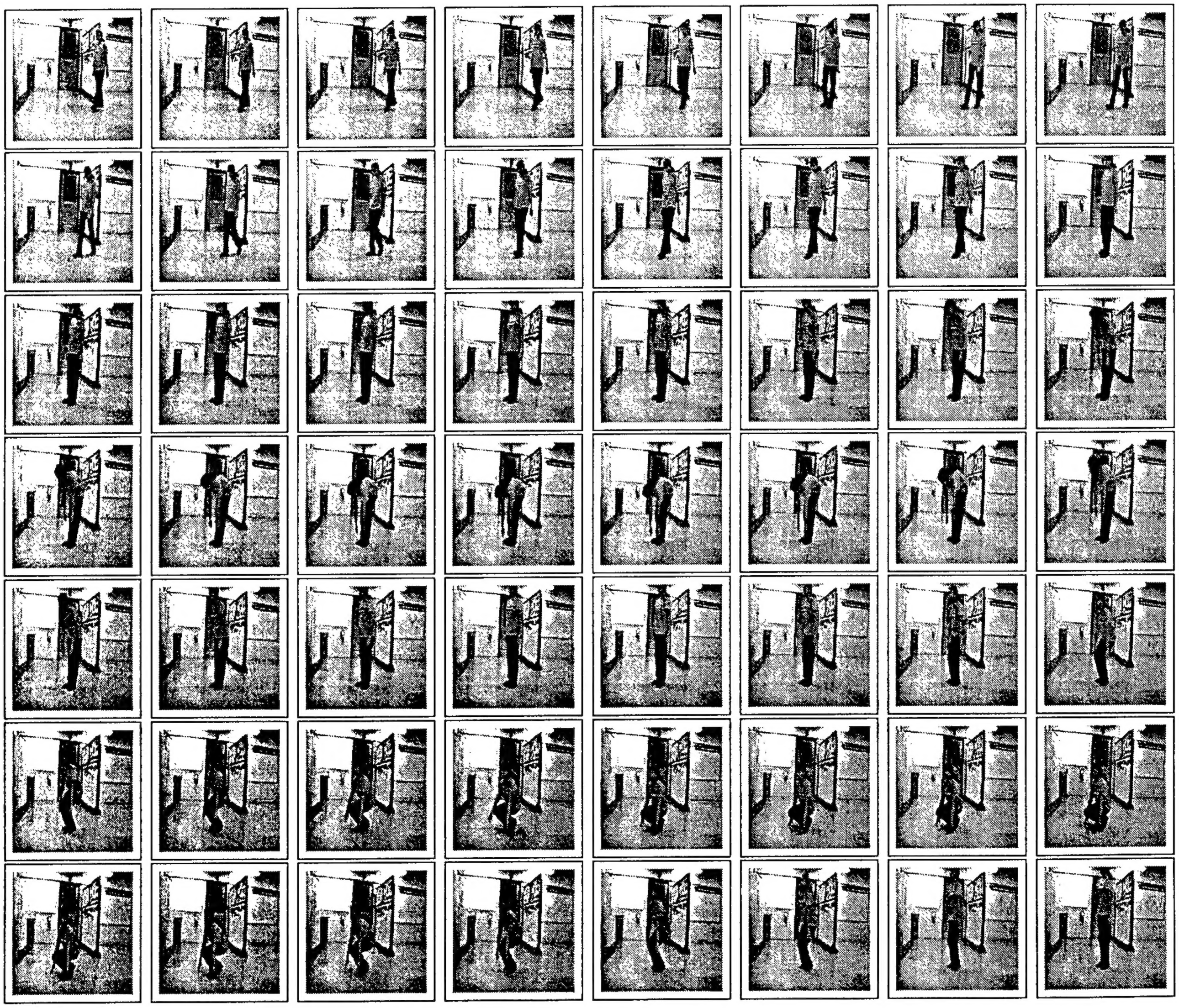

Figure 14: Frames 1 to 56 of test sequence 2

Test Sequence 2					
Actual frames	1-24	25-31	32-37	40-46	47-56
action	walking	bending	gettingup	squatting	rising
Detected frames	1-25	26-31	32-36	42-46	47-56
action	walking	bending	gettingup	squatting	rising

Table 3: Results for test sequence 2 taken at θ_1= 40 degrees